An Efficient Two-Phase Model for Computing Influential Nodes in Social Networks Using Social Actions

Article *in* Journal of Computer Science and Technology · March 2018

DOI: 10.1007/s11390-018-1820-9

CITATION
1

READS
44

2 authors:

Mehdi Azaouzi
University of Sousse
3 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Lotfi Ben Romdhane
ISITCom, University of Sousse
53 PUBLICATIONS **233** CITATIONS
SEE PROFILE

Azaouzi M, Ben Romdhane L. An efficient two-phase model for computing influential nodes in social networks using social actions. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 33(2): 286–304 Mar. 2018. DOI 10.1007/s11390-018-1820-9

An Efficient Two-Phase Model for Computing Influential Nodes in Social Networks Using Social Actions

Mehdi Azaouzi and Lotfi Ben Romdhane

Modeling of Automated Reasoning Systems Research Laboratory LR17ES05
Higher Institute of Computer Science and Telecom, University of Sousse, Sousse 526-4002, Tunisia

E-mail: mehdi.azaouzi@gmail.com; lotfi.BenRomdhane@isitc.u-sousse.tn

Received December 13, 2016; revised October 14, 2017.

Abstract The measurement of influence in social networks has received a lot of attention in the data mining community. Influence maximization refers to the process of finding influential users who make the most of information or product adoption. In real settings, the influence of a user in a social network can be modeled by the set of actions (e.g., "like", "share", "retweet", "comment") performed by other users of the network on his/her publications. To the best of our knowledge, all proposed models in the literature treat these actions equally. However, it is obvious that a "like" of a publication means less influence than a "share" of the same publication. This suggests that each action has its own level of influence (or importance). In this paper, we propose a model (called Social Action-Based Influence Maximization Model, SAIM) for influence maximization in social networks. In SAIM, actions are not considered equally in measuring the "influence power" of an individual, and it is composed of two major steps. In the first step, we compute the influence power of each individual in the social network. This influence power is computed from user actions using PageRank. At the end of this step, we get a weighted social network in which each node is labeled by its influence power. In the second step of SAIM, we compute an optimal set of influential nodes using a new concept named "influence-BFS tree". Experiments conducted on large-scale real-world and synthetic social networks reveal the good performance of our model SAIM in computing, in acceptable time scales, a minimal set of influential nodes allowing the maximum spreading of information.

Keywords social network, social influence, social action, personalized PageRank, influence-BFS tree

1 Introduction and Related Work

The popularity of social networks has increased over the years, such as Facebook, Google+, Twitter and MicroBlog. These networks have been used to model the human friendship by a graph wherein the members, or their profiles, are modeled by nodes and each edge represents a relationship. Members can share experiences, express their views, and discuss ideas with their friends. Thus, some ideas can spread in a major way in the network. These ideas are published by members who have a high capacity of interactivity and a significant breadth of information dissemination. These members are called influential individuals and go by many names in the literature as experts[1], authorities[2], and leaders[3]. Detecting influential members is a critical issue in social network analysis, which has many applications, such as viral marketing[4-5], healthcare[6], and sensor placement[7]. Indeed, such influential users can be used in politics to alter political preferences in a community, or in marketing to advertise products to increase sales[5,8]. From an abstract point of view, influential users are those who, when "activated", can activate the maximum number of individuals in the social network. More precisely, influence maximization (IM) consists in computing a small number k of influential users (referred to as seeds) that influence the maximal number of users (referred to as influence spread)[9]. The proposed models in the literature for solving the IM problem can be broadly divided into two categories. In the first category, IM is considered as an algorithmic problem[9]; whereas in the second category, it is considered as a discrete optimization problem[8]. Unfortunately, it was shown that solving IM is NP-hard under both the independent cascade (IC)[10-14] and the linear threshold (LT)[10,15-17] models. For this, several approximation algorithms were proposed in the liter-

Regular Paper

ature[18-20]. For example, Wang *et al.*[18] proposed a community-based greedy solution to the IM problem. In [19], Jiang *et al.* introduced a local approximation, called the expected diffusion value (EDV), to approximate the influence spread in the IC model computed by the Monte Carlo simulations. Tang *et al.*[20] proposed new randomized approximation algorithms which are orders of magnitude faster than the original greedy algorithms in [8]. Doubtless, an extensive review of existing models for solving the IM problem goes beyond the scope of the current paper and the interested reader is referred to the specialized literature, for example, [21-23]. Instead, hereafter, we will try to briefly categorize the existing approaches based on the adopted methodology.

In the first category, users are ordered w.r.t. a defined measure for influence and the top-k influential ones are selected. Starting with [3], a random-walk-based algorithm LeaderRank was proposed to identify leaders in social networks and outperforms PageRank. Weng *et al.*[24] defined a metric similar to PageRank for topic-sensitive influential users' detection in Twitter. The influence of a twitterer is computed by taking into account both the topical similarity between users and the link structure. In [25], Barbieri *et al.* introduced novel topic-aware influence-driven propagation models. Similar to traditional PageRank, the weight of each user is computed by a random walker with different probabilities. Xiang *et al.*[26] proposed a linear social influence model which is essentially PageRank with priors. To better qualify an individual, recently Wang *et al.*[27] have defined two distinct properties, namely, "susceptibility of being influenced" and the "influential power" in the PPRank algorithm. In [28], a fine-grained feature-based social influence (FBI) evaluation model is proposed. A user's initial social influence is computed based on two factors: the possibility of impacting others and the importance of the user him/herself. Then, the social influence of an individual is adjusted with PageRank taking into account the social influence contributions of his/her friends. Most of the above studies can be considered as heuristic approaches, since they explore the top-k influential nodes selected according to different types of centrality measures (such as the personal characters, PageRank, closeness, betweenness and topics). An alternative strategy always uses centrality measures to quantify the involvement of users. In fact, it is believed that people having the greatest number of links (connections) are the hubs for extensive influence spreading[29-30]. One of the most important measures is the interaction behaviours. For example, Li *et al.*[31] proposed CINEMA (Conformity-Aware Influence-Maximization), a novel conformity-aware cascade model for estimating influence spreads. CINEMA computes the influence and conformity indices of nodes by leveraging CASINO (conformity-aware social influence computation)[32]. In the algorithm CASINO, each edge is annotated with a positive or a negative sign according to the 5-leveled sentiments. Li *et al.*[33] studied voter-model dynamics on signed digraphs and applied it to solve the influence maximization problem. He *et al.*[34] proposed a greedy algorithm to address the positive opinion influential node set selection problem by considering both friend and foe relationships. Guler *et al.*[35] incorporated the social and physical network dynamics (such as propagation delay, frequency of interaction, the strength of friendship/foe ties or the impact factor of the propagating idea) to study the optimal influence propagation. Liu *et al.* proposed HYDRA[36-37] to address the IM problem across multiple social media platforms. In HYDRA, the authors of [36-37] combined temporal network information and node labels for similarity estimation. Extensive experiments on 10 million users across seven popular social network platforms demonstrate that HYDRA correctly identifies real user linkage across different platforms, and outperforms existing state-of-the-art algorithms by at least 20% under different settings, and four times better in most settings[36].

In the second category of approaches, the IM problem is modeled as an "individual influence evaluation problem". In [38], Subbian *et al.* used the social capital values to find influencers in online social networks. Franks *et al.*[39] proposed a multi-agent system to identify influential agents by adopting a matrix factorization method. Recently, Deng *et al.*[40] proposed an approach that incorporates node features and leverages the temporal nature of influence for influence maximization. Liu *et al.*[41] proposed a trust-oriented social influence evaluation method, called TOSI, that takes the social contexts into account. More recently, Zeng *et al.*[42] considered a new type of influence maximization problem which is naturally motivated by the reliability constraint of nodes in social networks. In [43], the problem of influence analysis is introduced in the context of information flow in networks. Then, a new algorithm InFlowMine is proposed based on a fully content-centered model of flow analysis in networks. In InFlowMine, the analysis is based on the actual content transmissions in the underlying social stream, rather than a static

model of transmission on the edges. In [44], Liu *et al.* proposed a model in which vertex attributes are considered in the IM problem. This proposal is motivated by the fact that vertices have different attributes in mobile social networks. For this, the authors of [44] suggested that it is more useful to study the IM problem in different search categories. They named this new problem as categorical influence maximization (CIM). The CIM query finds a small subset of vertices with different labels (categories) having the maximum influence spreads. To address CIM, the authors of [44] proposed a probability distribution based search (PDS) method mainly in two steps. In the first step, they proposed a probability distribution based parameter free method (PD-max) to identify the maximum influential vertex set for the specified category by studying the categorical influential distribution within a time interval. In the second step, starting from these detected influential vertices, they computed the minimum number of vertices in each category having the maximum influences using a probability distribution based minimizing method (PD-minmax). Experimental results on real-world datasets collected in a city of China during one year period used by the authors of [44] show the effectiveness of the proposed approach. In the same context, the graph summarization has also been utilized for information diffusion in dynamic graphs[45].

In the third category, we measure the "engagement" capacity of users in social networks, while we define the engagement as the capacity of a user in interacting with other users of the social network. First, we should notice that little research has been done on measuring engagement so far. In [46], the term of engagement is introduced in modeling email message chains. In [47], Achananuparp *et al.* proposed a study on Twitter where they introduced a new measure of engagement based on the count of re-tweets. We should remark that in these proposals, only direct responses from one user to another are accounted for, and that "intermediate" responses (engaging one user through another) are not considered. Zhao *et al.*[48] explored how the structure of online interactions affects the sentiment of the messages posted by a given user, considering the situations where the same user appears at least twice in the same thread (for example, asking a question and then replying to the respondents). In [49], Yang and Tang explored the reasons why a user may succeed in attracting responses to his/her posts, to understand the mechanisms of online influence. More recently, Nikolaev *et al.*[50] introduced a new metric (named the engagement capacity) for measuring the ability of online media platform users to engage each other in communication. In their work, the engagement capacity of a user is measured according to the number of newly contributed posts that attract more posts. The authors of [50] adopted a game theoretic approach to quantify engagement, view a platform's social capital as a cooperatively created value, and find a fair distribution of this value among the contributors. Extensive experimentation shows that engagement capacity can reveal well-interpretable facts about the nature of online communication on different platform types. The engagement capacity distribution in a userbase reveals the different dynamics of communication and engagement in two social media, differing in purposes, Health Forum and Twitter social networks.

We should notice that most of the proposed approaches solve the IM problem under the assumption that each user has a fixed cost for being an initial adopter. However, in practice, user decisions regarding the fact to be initial adopters or not are often probabilistic. Moreover, these models solve the influence maximization problem given a limited budget k, i.e., they accept k as a parameter. Hence, computing the optimal budget k becomes problematic and is often solved experimentally through a trial-and-error approach. In this regard, it is necessary to understand how users make reactions when they are influenced by a product, for adopting the allocated budget of influence. Moreover, to the best of our knowledge, all proposed models in the literature treat these reactions equally. In fact, such models use objective functions based on the number of interactions independently of their nature. However, in real life, people can actually react to the published content of their neighbors in several ways based on profound impact on them. The strength of the reaction differs from one publication to another. For example, in Twitter, having a "retweet" is not the same as having the mention "favorite". In Facebook, a "like" of a publication means less influence than a "share" of the same publication. This suggests that each action has its own level of influence (or importance).

In this paper, we propose a model called SAIM (Social Action-Based Influence Maximization) that falls in the first category. SAIM identifies influential users in social networks based on their interaction behaviors. To intertwine the IM with reality, we use a particular form of corresponding probability. Unlike other models, SAIM is not designed to work on the canonical information diffusion graph, which includes the modeling of

influence probabilities between nodes; rather it leverages on social actions between users and on the concept of node-centric influence power. The key feature of SAIM is that it does not require the seed size k, but rather computes an optimal one for the social network at hand. In addition, we propose a new measure of influence based on social actions that aims at computing the influence power for each user and in which actions are not treated equally. The remainder of this paper is structured as follows. Section 2 introduces preliminary material. In Section 3, we present the details of our proposal, illustrate it on a sample network, and analyze its time and space complexities. Section 4 conducts an extensive experimentation of SAIM and compares it with other recent proposals using large-scale real-world and synthetic social networks. Section 5 offers concluding remarks and sheds the light on future research directions.

2 Preliminaries

In this section, we first present the traditional influence maximization (IM) problem in social networks, and thereafter introduce our definition of the IM problem based on social actions.

2.1 Problem Description

The influence maximization problem was first defined by Kempe *et al.*[8] as follows. Given a network $G = (V, E)$, a diffusion model, the influence maximization problem aims to find a subset S of k nodes ($|S| = k$) such that the expected number of overall activated nodes $\sigma(S)$ is maximized, i.e.,

$$S^* = \underset{S}{\operatorname{argmax}}\, \sigma(S). \tag{1}$$

Kempe *et al.*[8] proved that the influence maximization problem defined in (1) is NP-hard and that the objective function $\sigma(S)$ is submodular under both the Independent Cascade (IC)[10-14] and the Linear Threshold (LT)[10,15-17] models.

In the age of social networks evolution, analyzing social interactions is an important criterion to identify the active members. In fact, in the study of influence maximization, either ignoring the social actions in the relationship polarities between users, or treating them incorrectly, will have a bad effect in practical applications[51].

To address this issue, we will present our definition of the IM problem using social actions. Our social-actions influence maximization problem focuses on the introduction of the activities of members to identify the most active users in the network. Doubtless, we will present the corresponding cascade model that analyzes the information diffusion. Formal definition follows.

Definition 1 (Social Actions Based Influence Maximization Problem). *Given a social network, $G = (V, E, A)$, where V is the set of nodes modeling the users of the network, E is the set of edges, and A is the set of social actions. We classify the nodes into two subsets: an active set B, and an inactive set $V \backslash B$. This classification is based on the social actions A. Thereafter, we find the minimum most active node set INF in B and a specific cascade model CM, such that the expected number of nodes influenced by INF according to CM is the largest in V. That is*:

$$S = \underset{INF \subseteq B, |INF|_{\min}}{\operatorname{argmax}} \sigma(INF). \tag{2}$$

Observe that in the aforementioned definition, we have introduced the notion of active member. Naturally, the user activity can be modeled through the reactions he/she receives from his/her friends. In fact, the probability of a node v to be influential can be viewed as a function of the social actions that v received from his/her friends. Likewise, we seek a theoretical analysis of how an arbitrary node's influence spread quality is affected. Given a set of active nodes INF and a propagation model CM, we seek to find a minimum number of active nodes that maximize $\sigma(INF)$ according to CM. After introducing our definition of the IM problem, now we need the following basic definitions.

2.2 Basic Definitions

We consider a directed graph $G = (V, E)$ with $V = \{v_1, v_2, \ldots, v_n\}$ being the set of vertices and the set of edges $E = \{(v_i, v_j) |$ there is an edge from v_i to $v_j\}$. A node represents an individual, and an edge between two nodes represents some kind of relationship (friendship or co-authorship, etc.). We denote by $|.|$ the set cardinality. Each vertex $v \in V$ in G is labeled with its computed influence score noted by $IP(v)$. Now, we are ready to introduce the following definitions.

Definition 2 (Direct Neighbor). *In $G = (V, E)$, vertex v is a direct neighbor of vertex u if v and u are connected by an edge. This relationship is represented by the edge $(u, v) \in E$.*

Definition 3 (Vertex Border). *In $G = (V, E)$, $B(u)$ is the set of all direct neighbors of vertex u, i.e., $B(u) = \{v \in V, (u, v) \in E\}$.*

Given these basic definitions, now we are ready to introduce our proposal for influence maximization subsequently.

3 Our Proposal

Our model, called SAIM, is composed of two major steps and is depicted in Fig.1. The main objective of the first phase is to compute the influence power of each node in the social network. Intuitively, this influence power is computed from users' actions on the published statutes by friends (nodes) at hand. Thereafter, SAIM prunes in the second phase the insignificant nodes using local average influence power, based on an assumption of convexity which leads to stopping conditions with respect to the distance that defines "locality". Thus, SAIM derives the set of influential nodes using a new concept named "influence-BFS tree". Subsection 3.1 outlines our proposal for computing the influence power, i.e., the first phase of SAIM.

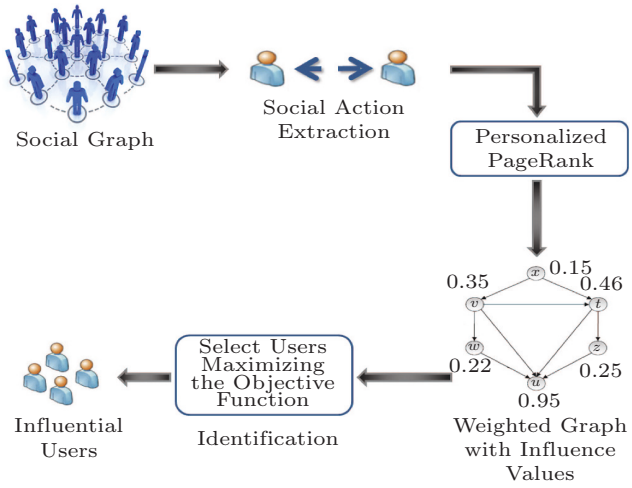


Fig.1. Workflow of the proposed approach.

3.1 Influence Power Model

Recent studies have introduced several factors for measuring influential power such as network properties and structural characteristics[29], topics[25,52-53], communities[17,54], quality of followers, quality of publications[55], and interaction behaviors[33-35,56-57]. Among the factors that directly measure the power of influence is the set of interactive behaviors. Generally, when a user is influenced by another, it will react accordingly. This reaction is modeled by a social action such as like, comment or share. The strength of the reaction indicates the strength of the influence or "influence power" (IP).

Generally, most people share the published content (publications) of friends considered to be useful or new. Indeed, they could attract more audience and amplify influence quickly by publishing original and high quality publications frequently. Thus, the interactions a user receives after each published content are an important indicator to measure its influence power. To the best of our knowledge, existing models consider only the number of social actions in the set of interactive behaviors. However, not all actions are of equivalent importance. Stated otherwise, they do not all have the same "level" of influence power. Hence, when a user v is influenced by the published content of user u, then the former will make a reaction depending on the influence power of such publication. For example, it is well-known in a social network like Facebook that a "share" is a much more important reaction (thereby meaning more influence) than a "like". Recently, Facebook has introduced even distinct levels of the social action "like" modeling distinct degrees of influence①. Hence, it seems natural to take into account this concept in computing the influence power of each individual in the social network.

In our model SAIM, we associate importance values, called friendship factors, to social actions. For this, let $A = \{a_1, ..., a_n\}$ be a finite set of social actions, and $\alpha = \{\alpha_1, ..., \alpha_n\}$ the set of friendship factors such as, $\forall i \in \{1, ..., n\}$, $\alpha_i > \alpha_{i+1}$ and $\sum_{\forall i} \alpha_i = 1$. Therefore, the influential power $W(u_x, u_y)$ of user u_y on his/her friend u_x can be described as follows:

$$W(u_x, u_y) = \frac{\sum_{i=1}^{n} \alpha_i \times N_{a_i}(u_x, u_y)}{N_{p_y}}, \quad (3)$$

where N_{p_y} is the number of published contents by user u_y, $N_{a_i}(u_x, u_y)$ is the number of actions a_i performed by user u_x on the published content of u_y (see Fig.2).

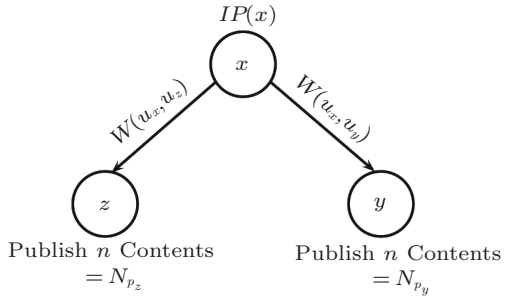


Fig.2. Transition influence power computation.

①https://newsroom.fb.com/news/2016/02/reactions-now-available-globally/, Feb. 2018.

Bonchi *et al.*[51] summarized several aspects in social network analysis such as trust, expertise and information propagation. They concluded that "the idea of influence in social networks is rather straightforward: when users see their social contacts performing an action they may decide to perform the action themselves". Based on this aspect, the users are interested in the publication that is preferred by their neighbors or is similar to their favorite publications. Based on measurements of the received endorsement, we can estimate the influence power of an individual. Intuitively, there are a large number of "endorsement paths" between the target user and the rest. Therefore, some form of proximity measure is required to compare the number of "endorsement paths". A general approach for measuring nodes' proximity in a network is personalized PageRank (PPR)[58] that is acknowledged to be one of the most effective measures that rank nodes based on their reachability from a certain set of nodes in a network. It gives high scores to items that are closer to the target user regarding a wide range of graph properties such as the distance or number of paths between them[59]. For this, PPR can be used to find the influence closeness of a node based on measurements of its endorsement. More clearly, based on users' behavior, PPR estimates the probability that a random walker, starting from the target user, will follow a path to influential and uninfluential users. For this, we adopt the PageRank algorithm to compute the influence based on our measurements of endorsement defined in (3). Hence, we propose the following PageRank-like function:

$$IP(u_x) = d \times \left(\sum_{u_y \in Followers(u_x)} \frac{W(u_y, u_x) \times IP(u_y)}{Followees(u_y)} \right) + (1-d)\frac{|Followers(u_x)|}{N},$$

where d is a dumping factor.

Algorithm 1 computes the influence power (IP) for each vertex in the input graph. Fig.3 depicts a sample graph in which each vertex is labeled with its computed IP value. Now, we need to compute a subset of nodes which will be the final seeds. This process is described in Subsection 3.2.

3.2 Significant Nodes Generation

After the computation of the IP values in the network, this phase aims to determine a set of candidate seeds based on the influence score of each node and its connectivity in the network. Notice that as social networks in realistic settings are extremely large, the search space for selecting seeds with maximal influence spread is also huge. Therefore, there is a need to effectively reduce the number of candidate seeds. How to narrow down the size of the candidate set of seeds is a core issue in our model SAIM tackled in this phase.

Based on our observations, an individual (say, v_x) with a high influence power is trusted by his/her friends, and therefore triggers more friends (friends-of-friends) to follow him/her. Hence, starting from v_x, this influence is propagated through the network following distinct paths composed of friends and friends-of-friends. Naturally, this influence power decays as we move from v_x until it is completely annihilated. Hence, this measure of influence power defines an influence zone for v_x. Naturally, an intuitive approach would select such centroids having the k-high influence

Algorithm 1. Influence Power

Data: a graph $G = (V, E)$, a social action set $A = \{a_1, ..., a_n\}$, a set of friendship factors $\alpha = \{\alpha_1, ..., \alpha_n\}$
Result: an influence value of every vertex $u \in V$

```
for (u_i, u_j) ∈ E do
    for a_x ∈ A do
        Calculate α_x × N_{a_x}(u_i, u_j)
    end for
    Calculate W(u_i, u_j)
end for
for u_i ∈ V do
    F(u_i) = |Followers(u_i)|/N
end for
for u_i ∈ V do
    for u_j ∈ Followers(u_i) do
        sum = sum + W(u_j, u_i) × IP(u_j)/Followees(u_j)
    end for
    IP(u_i) = (1 − d) × F(u_j) + d × sum
end for
```

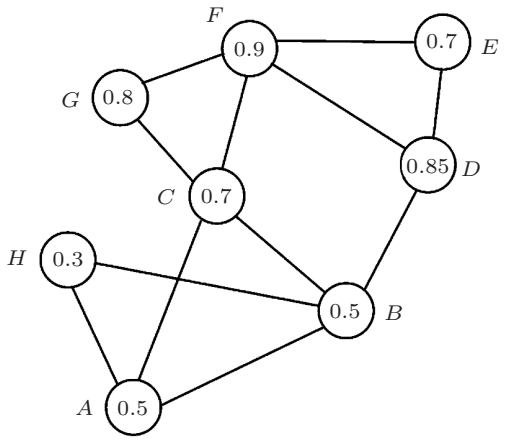


Fig.3. Sample graph where each node is labeled with its IP value computed by PPR.

scores as the k-influential seeds. For this, we define a significant node as a node with an influence power (IP) greater than the average IP in a given zone of the social network. By pruning the insignificant nodes, we can effectively reduce the number of seed candidates. In order to find the significant nodes, we use the local average value of IP for each vertex. In this model, only directed paths are considered. Let $P = (v_1, v_2, ..., v_m)$ be a shortest path which leads v_1 towards v_m. The length of P is its number of edges. Now, we can define the local average influence as follows.

Definition 4 (Local Average Influence). *Given a graph $G = (V, E)$ where each vertex v is labeled with its IP value, we define the local average influence value of v as follows*:

$$I_L(v) = \frac{1}{N} \sum_{v' \in path(v,L)} IP(v'),$$

where N is the number of nodes in all the shortest paths with length L from v.

In Definition 4, the notion of locality of $I_L(v)$ is related to the radius L of the region in the social network centered around vertex v.

Definition 5 (Significant Vertex). *Given a graph $G = (V, E)$ where each vertex v is labeled with its IP value, a node v is said significant iff: $IP(v) > I_L(v)$.*

A key challenge in our definition of a significant vertex is what is the best length L to be used. To solve this problem, we can estimate the length L by determining optimal local influence of any vertex v based on these properties.

Definition 6 (Convex Function). *A function f is said convex*[60] *if $\forall x, y \in \mathrm{dom}(f)$ and $\lambda \in [0, 1]$, $f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda)f(y)$. Moreover, let a set C be convex, if for all points x, $y \in C$ and $0 \leqslant \lambda \leqslant 1$ we have $\lambda x + (1-\lambda)y \in C$.*

Theorem 1. *$I_{L(v)}$ is a convex function.*

Proof.

$$\begin{aligned} I_{L'}(v) &= \frac{IP_L(v) + IP_{L'\setminus L}(v)}{N_L + N_{L'\setminus L}} \\ &= \frac{N_L}{N_L + N_{L'\setminus L}} I_L(v) + \frac{N_{L'\setminus L}}{N_L + N_{L'\setminus L}} I_{L'\setminus L}(v). \end{aligned}$$

Let $\lambda_1 = \dfrac{N_L}{N_L + N_{L'\setminus L}}$ and $\lambda_2 = \dfrac{N_{L'\setminus L}}{N_L + N_{L'\setminus L}}$, then

$$\begin{aligned} I_{L'}(v) &\in \mathrm{convex}(I_L(v), I_{L'\setminus L}(v)), \\ &= \{\lambda_1 I_L(v) + \lambda_2 I_{L'\setminus L}(v), \end{aligned}$$

such that $\lambda_1 + \lambda_2 = 1, 0 \leqslant \lambda_1 \leqslant 1\}$. □

Now, we can derive the following properties.

Property 1. $\{L | I_L(v) > I_{(L+1)\setminus L}(v)\} = \emptyset \Rightarrow \forall L, I_L(v) \leqslant I_{(L+1)\setminus L}(v)$.

Property 2. $\{L | I_L(v) > I_{(L+1)\setminus L}(v)\} \neq \emptyset \Rightarrow L_0 = \min\{L | I_L(v) \leqslant I_{(L+1)\setminus L}(v)\}$.

The last two properties give us stopping conditions when computing the optimal length L using a greedy algorithm. Indeed, as long as the average influence increases and becomes greater than the influence power of the central node, then we can decide that such node is not significant. On the contrary, if the local average influence decreases, then the optimal length is the minimal one causing that decrease. Now, we are ready to introduce the following proposition.

Proposition 1 (Local Length of a Vertex). *Given a graph $G = (V, E)$ wherein each vertice v is labeled with IP value, for a given vertex v, we define the set of length $\Omega(v)$ where $I(v)$ is ascending, and is given by*:

$$\Omega(v) = \{L, I_L(v) > I_{(L+1)\setminus L}(v)\}.$$

$\Omega(v)$ is bounded below 1. Now, we define the local length of vertex v by the minimum length in $\Omega(v)$, $L_0 = \min(\Omega(v))$.

Algorithm 2 outlines our computation of the optimal length. First, a path of length $l = 1$ is considered and $I_l(v)$ is computed. Stated otherwise, first we consider only the direct neighbors of each vertex. If $IP(v)$ is greater than $I_l(v)$, then the path length is increased by 1 and $I_l(v)$ is computed again. Here, we should consider two cases. In the first case, the local average value $I_l(v)$ is decreased compared with the previous path length $I_{l'}(v)$. Hence, $IP(v)$ remains higher than $I_l(v)$ and vertex v is noted to be significant. In the second case, the local average value $I_l(v)$ is increased

Algorithm 2 . Significant Nodes

Data: a weighted graph $G = (V, E, W)$ where each node is labeled with its influence score
Result: a set of significant nodes $B = \{v_1, ..., v_p\}$

```
1: B ← ∅
2: for i ← 1 to |V| do
3:     L ← 1
4:     Compute I_L(v_i)
5:     while (I_{L\(L+1)}(v_i) > I_{L+1}(v_i)
        and IP(v_i) > I_L(v_i)) do
6:         L ← L + 1
7:     end while
8:     if IP(v_i) > I_L(v_i) then
9:         B ← B ∪ {v_i}
10:    end if
11: end for
12: return B
```

compared with the previous path length, then the path length is increased, and $I_l(v)$ is computed again. Consequently, if $IP(v)$ becomes lower than $I_l(v)$, then we should stop our expansion of the region and decide that the node v is not significant.

Definition 7 (Significant Vertices). *Given a graph $G = (V, E)$ where each vertex v is labeled with IP value, we define the set of significant vertices of G as follows*:

$$B = \{v : v \in V \text{ and } IP(v) > I_{L_0}(v)\}.$$

Given a social network $G = (V, E, B)$, let B be the set of significant nodes computed by Algorithm 2. Hence, $V \backslash B$ is the set of non-significant ones. For the sake of presentation, we will mark the significant nodes as black, whereas the non-significant ones as white. Fig.4 reports our sample graph (in Fig.3) on which we have run Algorithm 2. We can remark that the set of computed significant (black) nodes is $B = \{$"A", "D", "F", "G"$\}$, whereas the rest of the nodes are labeled as non-significant (i.e., those in white). Basically, our model SAIM specifies two activities to represent the behavior of the individuals in G when faced with a given information (publication): 1) a black node who is "infected" by the information may keep sending it to its neighbors, and 2) a white node can be influenced by this information but it cannot keep sending it to its neighbors. This means that black nodes receive and send influence, where white nodes are only influenced. Now, we need to compute an optimal subset of these computed black nodes which would maximize influence as we will see in Subsection 3.3.

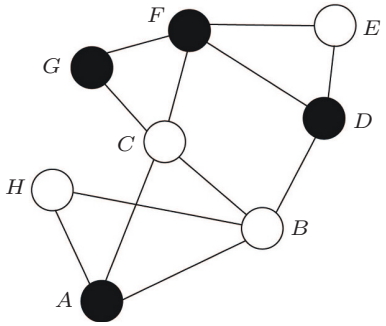


Fig.4. Sample graph in which black nodes represent the significant ones and the white nodes represent the non-significant ones.

3.3 Finding Seed Nodes

As mentioned previously, we want to identify an optimal subset of black nodes as seeds, i.e., those that maximize the number of influenced nodes at the end of the diffusion process. For this, we will introduce a new concept called "influence-BFS tree", for pruning candidate seeds.

3.3.1 Influence-BFS Tree

Breadth first search (BFS) is a strategy for traversing graphs. We will base our proposal on this strategy to solve the influence maximization problem. The rationale behind the choice of BFS is that it produces the shortest paths (in terms of the number of edges) and thereby guarantees a rapid spread of information. Hence, given a black (or significant) node u in the social network, we can compute the set of influenced nodes (black or white) by u with a specific BFS tree rooted at u. Our methodology for computing influence-BFS trees, called influence-BFS, is a variant of the standard breadth-first search technique, which imposes constraints on the visited nodes as follows. At each step, influence-BFS starts with a black vertex and puts it in an empty queue. Then, the first vertex is extracted from the queue and all its unvisited neighbors are visited and added to the queue. The main difference between the standard BFS algorithm and the influence-BFS is that in the latter only black vertices are selected to build the queue for the next level. This choice is natural since only significant black nodes can diffuse or transmit information and thereby trigger friends. For each vertex, its distance from the root or its predecessor is stored in an array called distance (parent array) that represents the output of the algorithm. For each vertex in the current level all its neighbors must be visited. For example, Fig.5 reports our sample graph $G = (V, E, B, W)$ and four different influence-BFS trees (T_1, T_2, T_3 and T_4) for G. For example, in T_1, BFS begins by the black vertex F, then all its black and white neighbors $\{C, E, D, G\}$ will

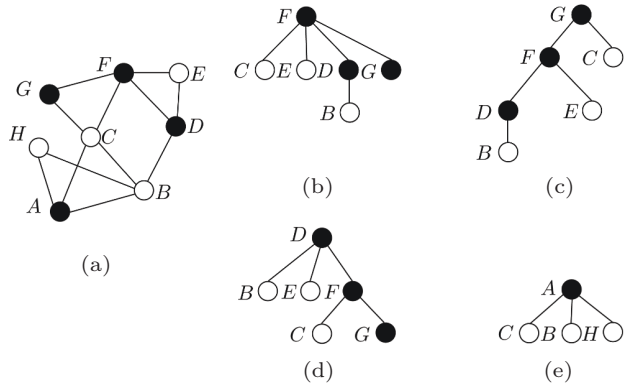


Fig.5. Input graph G and four corresponding influence-BFS trees. (a) Input graph G. (b) T_1. (c) T_2. (d) T_3. (e) T_4.

be visited. The addition at the end of the queue is allowed only on black vertices D and G. Finally, the unvisited neighbors of D and G (i.e., vertex B) will be visited.

Now, we need the following definitions and properties related to the concept of influence-BFS tree.

Definition 8 (Black Path). *Let us consider a graph $G = (V, E, B)$, where B denotes the set of black nodes. We define a black path $B_{\text{path}}(u, v)$ between two black nodes $u \in B$ and $v \in B$ as a sequence of edges that only connects a sequence of black nodes.*

For example, in Fig.5, $B_{\text{path}}(G, D) = (GF, FD)$ is a black path between black nodes G and D.

Notation 1. *Given $G = (V, E, B)$ and a vertex $u \in B$, we denote by T^u the influence-BFS tree of black node u.*

Since, we associate an influence-BFS tree to each black node, and then we can say that the number of influence-BFS trees in the social graph is equal to the number of black nodes. This is modeled by the following property.

Property 3. *Given $G = (V, E, B)$, where B is the set of black nodes, and $T = \{T_1^{v_1}, T_2^{v_2}, ..., T_k^{v_k}\}$ is the set of influence-BFS trees of G. Then, we have $|T| = |B|$.*

Definition 9 (Size of Influence-BFS Tree). *Given $G = (V, E, B)$, where $|B| = r$ and $T = \{T_1^{v_1}, T_2^{v_2}, ..., T_k^{v_k}\}$, the set of influence-BFS trees. We note by size$(T_i^{v_i})$ the size of $T_i^{v_i}$ defined as its number of nodes (in black and white).*

For example, in Fig.5, the size of $T_1^F = 6$, while the size of $T_4^A = 4$. Intuitively, the size of T^{v_i} answers the question how many nodes are influenced by the black (significant) node v_i, i.e., the influence spread. Naturally, the black vertex having the maximal influence-BFS tree is the most influential.

Definition 10 (Influence Zone). *Let $G = (V, E, B)$, where $|B| = k$ and $T = \{T_1^{v_1}, T_2^{v_2}, ..., T_k^{v_k}\}$, the set of influence-BFS trees. The influence zone of black vertex v_i is the set of nodes influenced by v_i, i.e., $A_{v_i} = \{u | u \in T_i^{v_i}\}$.*

At this stage, we should notice that our construction of the influence-BFS tree respects the following constraint: an internal node at level k is only developed (i.e., visited) from a given node at level $(k-1)$ which is necessarily a black (significant) node. This is defined formally as follows.

Property 4. *Let $G = (V, E, B)$, where B is the set of black nodes. Let T^u be the influence-BFS tree of $u \in B$, and let $x \in T^u$ be an internal node of it. Let $\pi(u, x) = (u, x)$ be the path between the root u and x. Then, π has the form $\pi(u, x) = (u, v_1, v_2, \cdots, v_k, x)$ where $v_1, v_2, \cdots, v_k \in B$.*

Lemma 1. *Consider a social graph $G = (V, E, B)$ and two vertices $u, v \in B$. There exists a black path between u and v, $B_{\text{path}}(u, v)$ if and only if $v \in T^u$.*

Proof. Let us assume that $v \in T^u$, and then there exists necessarily a black path between the root u and v due to Property 4.

Now, let us assume that there exists a black path between u and v, i.e., $\pi(u, v) = (u, x_1, x_2, \cdots, x_r, v)$ where $x_i \in B, \forall i$. Hence, running BFS starting from u will enable us to visit x_1 at level 1, x_2 at level 2,..., and x_r at level r which are all black nodes, and thereby are included in the influence-BFS tree T^u. Starting from x_k, we thereafter visit v at level $(k+1)$ whether it is a white or black vertex. Consequently, $v \in T^u$. □

Corollary 1. *Given $G = (V, E, B)$, where $|B| = k$ and the set of influence-BFS trees $T = \{T_1^{v_1}, T_2^{v_2}, ..., T_k^{v_k}\}$. Let A_{v_i} and A_{v_j} be the influence zones of both black vertices v_i and v_j. If there is a black path $B_{\text{path}}(v_i, v_j)$ between v_i and v_j then $A_{v_i} = A_{v_j}$.*

Corollary 1 simply states that the existence of a black path between two black nodes guarantees that their influential zones are identical. We can see for example in Fig.5 that there is a black path between nodes G and D. The influence zone of G is $A_G = \{G, F, C, D, E, B\}$ (i.e., influence-BFS tree T_2), whereas the influence zone of F is $A_F = \{F, C, E, D, G, B\}$ (i.e., influence-BFS tree T_1). It is obvious that $A_G = A_F$. Naturally, in order to reduce the search space, selecting one of these BFS trees is enough to guarantee the influence of the same zone. At this stage, a major concern raises about the choice of the best black vertex (or influence-BFS tree) among all possible ones in a black path. Naturally, the best tree would enable diffusion (or broadcast) of information very quickly.

Definition 11 (Rank-Vertex). *Consider an influence-BFS tree T^v and a vertex $u \in T^v$. The rank of u in T^v, denoted as $rank(u, T^v)$, is defined as the number of hops away from the root v to node u, $hop(v, u)$ i.e.,*

$$rank(u, T^v) = |\{hop(v, u) | u \in T^v\}|.$$

Having the rank of each node at hand, now we can define the rank of an influence-BFS tree as follows.

Definition 12 (Rank-Tree). *Consider an influence-BFS tree T^v. The rank of T^v is defined as the average rank of its vertices, i.e.,*

$$Rank(T^v) = \frac{1}{|A_v|} \sum_{u_i \in T^v} rank(u_i, T^v).$$

Definition 13 (Minimal Rank-Tree). *Consider a black path* $B_{\text{path}}(v_i, v_j)$ *and* $T = \{T_k^{v_k} | k \in B_{\text{path}}(v_i, v_j)\}$*, the set of influence-BFS trees of each vertex in* $B_{\text{path}}(v_i, v_j)$*. We call the minimal tree* $T_{\min}$*, the tree with the lowest rank, i.e.,*

$$T_{\min} = \text{argmin}\{Rank(T^{v_k}) | v_k \in B_{\text{path}}(v_i, v_j)\}.$$

Definition 13 states that the best BFS tree among those corresponding to root nodes in the same black path is the one that has, in the average, the shortest path to its nodes. Stated otherwise, it has the "fastest spread" of influence. As for our example in Fig.5, we consider the black path $B_{\text{path}}(G, D) = (G, F, D)$. The minimal rank tree in this black path is computed as follows in Table 1.

Table 1. Rank Tree Values for the Influence-BFS Trees in Fig.5

Tree	*Rank*
T^G	$\frac{rank(F,T^G)+rank(C,T^G)+rank(D,T^G)}{5} + \frac{rank(E,T^G)+rank(B,T^G)}{5} = \frac{1+1+2+2+3}{5} = 1.8$
T^F	$\frac{rank(C,T^F)+rank(E,T^F)+rank(D,T^F)}{5} + \frac{rank(G,T^F)+rank(B,T^F)}{5} = \frac{1+1+1+1+2}{5} = 1.2$
T^D	$\frac{rank(B,T^D)+rank(E,T^D)+rank(F,T^D)}{5} + \frac{rank(C,T^D)+rank(G,T^D)}{5} = \frac{1+1+1+2+2}{5} = 1.4$

As a summary, now we can say that each significant node is characterized by an influence-BFS tree which models its influence zone. In addition, we have proved that black nodes belonging to the same black path have the same influence zones. Hence, in order to prune the search space, we define a measure to rank those trees and choose the best (minimal) one. From an abstract point of view, the best tree is the one that spreads the information most rapidly: when this information is put on its root, it will reach most rapidly (in terms of path length) the rest of its nodes. Now, we are ready to outline our algorithm for computing the optimal set of influential nodes subsequently.

3.3.2 Algorithm

Our approach of influence maximization algorithm is summarized in Algorithm 3 and follows a greedy approach in computing the set of influential nodes. It takes as input a graph G where each vertex v is labeled with its influence score IP. The set of influential nodes INF is initially empty (line 1). First, we begin by computing B the set of black (significant) nodes in the graph using Algorithm 2 (line 2). Thereafter, we build the influence-BFS trees of nodes in B (lines 3∼6). The while loop (lines 7∼14) builds the influential nodes set INF from the black nodes B as follows. We select from B the node (say, $u_{\max}$) having the maximal influence spread (line 8) in graph G. Thereafter we compute the black path in the influence-BFS tree of $u_{\max}$ (line 9). Remind that all the black nodes in the black path BLACK have exactly the same influence zone and thereby are equivalent thanks to Corollary 1. This is why we select the node $v_{\min}$ corresponding to the influence-BFS tree $T_{\min}$ with the minimal rank. Stated otherwise, we select the BFS tree having the quickest broadcast of information from its root $v_{\min}$ to the rest of its nodes (see Definition 12). Hence, $v_{\min}$ is added to the actual set of influential nodes (line 12), whereas it is removed (along with all black nodes in the same path BLACK) from B (line 13). Finally, our algorithm outputs the selected nodes as seeds (line 15). As a summary, the seed selection is stopped when the maximal number of nodes in the network is influenced by the selected seeds. In addition, a new selected seed does not improve any further the expected number of activated nodes. Therefore, the optimal solution for the problem defined in (2) is obtained. In Subsection 3.3.3, we will illustrate the distinct steps of our algorithm on a sample social graph.

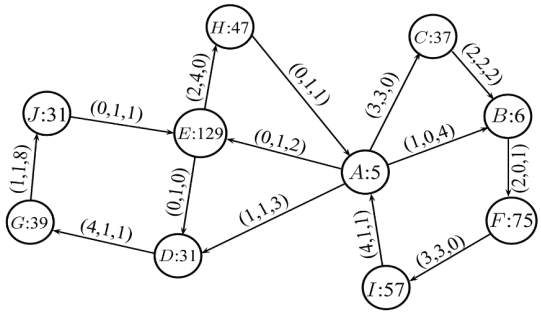
Algorithm 3 . Influence Maximization Algorithm

Data: a graph $G = (V, E)$, where each vertex is labeled by its influence power IP
Result: a set of influential nodes INF
1: $INF \leftarrow \emptyset$
2: $B \leftarrow$ Significant-Nodes(G)
3: **for each** $v_i \in B$ **do**
4: Build the influence-BFS tree $T_i^{v_i}$ of v_i
5: $T \leftarrow T \cup T_i^{v_i}$
6: **end for**
7: **while** $B \neq \emptyset$ **do**
8: $u_{\max} = \underset{v \in B}{\text{argmax}}\,(size\,(T^v))$
9: Compute the black path of $u_{\max}$: $BLACK \leftarrow \{v_k | v_k \in T^{u_{\max}} \wedge v_k \in B\}$
10: Let $T_{\min} \leftarrow \text{argmin}\{Rank(T^{v_k}) | v_k \in BLACK\}$
11: Let $v_{\min} \leftarrow$ the root node of $T_{\min}$
12: $INF \leftarrow INF \cup \{v_{\min}\}$
13: $B \leftarrow B \setminus \{BLACK \cup \{v_{\min}\}\}$
14: **end while**
15: **return** INF

3.3.3 Illustration

The main purpose of this subsection is to illustrate the distinct steps of our algorithm on a sample social graph composed of 10 nodes and 14 edges depicted in Fig.6. The graph is oriented: a link from u to v simply means that v performed an action on a publication of u. Each vertex (user) is labeled with his/her number of published contents, and an oriented edge (u, v) is labeled with the number of actions performed by user v on the publications of user u. For our concern, we consider only three social actions "like", "share" and "comment" (see Fig.6(a)). Moreover we associate the following social factors (importance) to these social actions empirically as follows: $\alpha_{\text{like}} = 0.15$, $\alpha_{\text{comment}} = 0.35$ and $\alpha_{\text{share}} = 0.50$. We should notice that $\alpha_{\text{share}} + \alpha_{\text{like}} + \alpha_{\text{comment}} = 1$ and that $\alpha_{\text{share}} > \alpha_{\text{comment}} > \alpha_{\text{like}}$. In fact, we consider that a share is more meaningful than a comment which is more meaningful than a like w.r.t. the reactions of users.

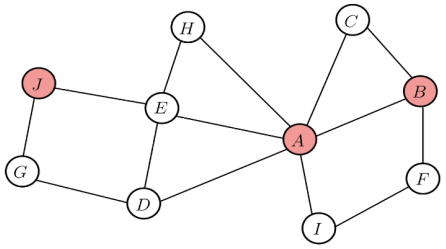
Initially we measure the endorsement weight from vertex u_x to vertex u_y which is subject to the probability function computed by (3). Then we compute the influence power IP of the vertices, by calling Algorithm 1 — see Fig.6(b). The set of significant black nodes is $\{A, B, J\}$ reported in Fig.6(c). The corresponding BFS trees are depicted in Fig.6(d). Remark that both black nodes A and B (having the maximum influence tree size) belong to the same black path. However, the influence-BFS tree of A has the lowest rank ($\approx$ 1.14). Hence, node A is chosen as the influential node along this path and is added to INF, and



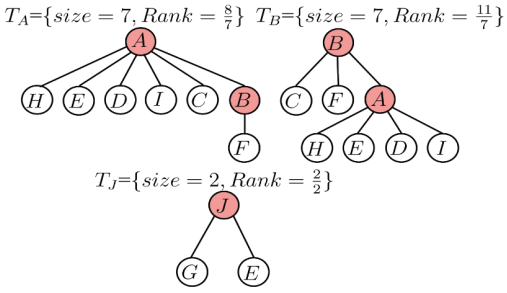
(a)

Vertice ID	Influence Power
A	0.209 823 857 041 132
B	0.208 989 368 213 113
C	0.151 807 604 134 553
D	0.153 603 159 033 322
E	0.151 413 744 476 988
F	0.151 894 837 000 891
G	0.154 854 253 755 158
H	0.152 327 583 581 467
I	0.153 397 647 574 827
J	0.169 107 016 224 549

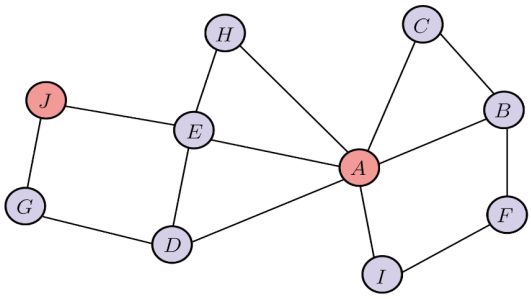
(b)



(c)



(d)



(e)

Fig. 6. Visualization of the main steps of our algorithm. (a) Sample graph with several social actions reported on its edges. (b) Computed influence power. (c) Significant nodes generation. (d) Influence-BFS tree set. (f) Seed's nodes.

both nodes A and B are removed from the set of black nodes which contains thereafter only J. Hence, J will compose the next seed and is removed from the set of black nodes which becomes empty, and therefore the algorithm stops. The final output of our algorithm is $INF = \{A, J\}$ as the set of influential nodes in this sample graph (see Fig.6(e)).

Most important aspects in the detection of influential users in large-scale social networks are both time and space complexities. For this, we will evaluate theoretically the performance of our algorithm by computing its temporal and spatial complexities subsequently.

3.3.4 Complexity Analysis

SAIM is run in three main steps: 1) computation of the influence power of each node using PPR, 2) generation of significant nodes, and 3) finding the seed nodes. Hence, we will evaluate the space and time complexities of each step and then sum up the results. The following theorem is about the time complexity.

Theorem 2. *The time complexity of SAIM is $O(n + m)$ where n is the number of vertices and m is the number of edges in the social network.*

Proof. We begin by the step of calculating the influence power (Algorithm 1), which is as follows. First, the computation of endorsement (lines 2∼7) is done in $O(m \times |A|) = O(m)$ where $|A|$ is the number of actions (generally constant w.r.t. the size of the networks) and m denotes the number of edges. Second, we calculate the followers of each vertex (lines 8∼10) in $O(2m) = O(m)$. Third, we calculate the IP value of each vertex (lines 8∼16) in $O(2m) = O(m)$. Therefore the time complexity of Algorithm 1 is: $T_{\text{algorithm 1}} = O(m \times |A| + m + m) = O(|A| + 2m)$. Since $|A|$ is a constant, $T_{\text{algorithm 1}} = O(m)$. After the computation of IP value of each vertex, we apply personalized PageRank (PPR) computation on these measures, which runs in the worst case p times with p being the number of iterations needed before personalized PageRank converges. The time complexity of the influence power model is:

$$T_{\text{step 1}} = O(p \times m).$$

Then we move to the second stage which is the computation of the local average influence (Algorithm 2). Calculating the area average influence of each vertex (lines 3∼13) is done in $O(n \times T)$ (T is the area size and it is $L_{\max}$ in the worst case, where $L_{\max}$ is the maximum length of a path in graph), thus its complexity is:

$$T_{\text{step 2}} = O(n \times L_{\max}).$$

We note that in the general case $L_{\max}$ is negligible compared with the number of nodes n and $L_{\max}$ is a constant. Therefore, the time complexity of this phase can be estimated to be $O(n)$.

In the third step, we apply the influence-BFS tree algorithm (Algorithm 3, lines 4∼7) on each black (significant) node. First, we begin by computing B (line 3), which has been already calculated, in $O(n \times L_{\max})$. The complexity of the BFS algorithm is $O(n')$, where n' denotes the number of nodes in the tree. Therefore, the time complexity of the influence-BFS tree is $O(n \times |B|)$. After, we select the seed nodes (lines 8∼15) which is $O(n)$. Therefore, the time complexity of this phase is:

$$\begin{aligned} &T_{\text{step 3}} \\ &= O(n \times |B| + n \times L_{\max}) = O(n \times (|B| + L_{\max})). \end{aligned}$$

Finally, we note that in the general case p and $|B| + L_{\max}$ are constants. Therefore, the time complexity of the entire algorithm can be estimated to be:

$$\begin{aligned} T_{\text{temporal}}(\text{SAIM}) &= O(p \times m) + O(n \times (|B| + L_{\max})) \\ &= O(p \times m + n \times (|B| + L_{\max})) \\ &= O(m + n). \end{aligned}$$

□

The following theorem is about the spatial complexity of SAIM.

Theorem 3. *The space complexity of SAIM is $O(m)$ where m is the number of edges in the social network.*

Proof. In our approach, we use two vectors: the first vector stores the influence power of nodes, and the second is the queue used for influence-BFS tree, which is stored at the worst case n nodes (all vertices). We also build a graph with m edges. Considering the worst case where the input graph is very dense (w.r.t. the number of edges), then we have:

$$T_{\text{spatial}}(\text{SAIM}) = O(\max\{m, n\}) = O(m).$$

□

In the next section, we will conduct an extensive experimental evaluation of our model SAIM on real world as well as synthetic social networks.

4 Experimentation

The main purpose of this section is to compare experimentally our proposal SAIM with the state-of-the-art research in the field of influence maximization. For this, we considered the following six well-known proposals: 1) maximum degree[8] which

is a heuristic based on the degree centrality measure, 2) CDH (Community and Degree Heuristic)[29] which is a model based on community detection, 3) CELF++ (Cost Effective Lazy Forward)[61] which is a fast and more efficient version of CELF[7], 4) SMG (State Machine Greedy)[62] which is a fast greedy algorithm, 5) CINEMA (Conformity-Aware Influence-Maximization)[31] which is a conformity-aware model based on the sentiment of users computed by the algorithm CASINO (Conformity-Aware Social Influence Computation)[32], and 6) IMM (Influence Maximization via Martingales)[20] which is a model based on the classic statistical tool martingales. As a testbed, we considered large-scale real-world and synthetic social networks. As evaluation criteria, we considered these standard measures: influence spread, coverage, and performance. At this stage, we should notice that CELF++, SMG, CDH, CINEMA and IMM require the edge propagation probabilities in the social networks. For this, we adopted exactly the same methods used in [20, 29, 31, 61-62] for generating edges probabilities. Hence, they were assigned as $b_{u,v} = 1/N^{\text{in}}(v)$ for all edges $(u, v) \in E$, where $N^{\text{in}}(v)$ is the set of nodes that has an edge to v. In all experiments, we set the parameters of HDM (Heat Diffusion Model)[29] which gives the best performance in CDH, where the activation threshold $\theta = 0.1$, flow duration $t = 0.1$, and thermal conductivity $\alpha = 0.1$ and we run 10 000 Monte Carlo simulations in CELF++. All experiments were implemented in the Java language, and were performed on a machine with Intel® Core™ I5 CPU at 3.20 GHz and 16 GB memory running Ubuntu Linux environment.

4.1 Evaluation on Real Datasets

In this first set of runs, we chose two widely used real datasets embedding the social actions. Both datasets are described in Table 2. The first public dataset is Tencent Weibo②. It is a sampled snapshot numbered in millions of users provided with rich information including demographics, profile keywords, follow history, interaction records, etc. In this dataset, three social actions are recorded, namely "retweet", "comment" and "tag". We associated, empirically the following social importance factors to these social actions as $\alpha_{\text{retweet}} = 0.50$, $\alpha_{\text{comment}} = 0.35$, and $\alpha_{\text{tag}} = 0.15$. Doubtless, these factors are ordered by the importance of the social actions. The second dataset is the Higgs Tweet available in SNAP③. It is extracted from Twitter between the 1st and the 7th of July 2012 on a specific topic. Note that this dataset has been updated on March 31, 2015. It includes three diffusion periods (before, during, and after the announcement) of the event. It includes three user activities in Twitter presented in the form of four directional networks. The user activities are "retweet", "reply" to existing tweets, and "mention" other users. We associated, empirically, the following social importance factors to these social actions as $\alpha_{\text{retweet}} = 0.50$, $\alpha_{\text{reply}} = 0.35$, and $\alpha_{\text{mention}} = 0.15$.

Table 2. Characteristics of the Real Networks

Dataset	Tencent Weibo	Higgs Tweet
Number of nodes	2 320 895.0	456 626.0
Number of edges	50 655 143.0	14 855 842.0
Maximum number of followers	456 827.0	67 502.0
Mean followers	55.0	67.0
Maximum number of followees	5 188.0	3 076.0
Mean tweets	113.3	102.4
Mean retweet	47.4	22.8
Mean comments	6.5	2.1
Mean At (mention)	12.3	10.1

Simulation results on these real-world datasets are reported in Table 3∼Table 8. For a better readability, in the simulation results outlined in this subsection, the best value(s) with respect to each criterion is(are) shown in bold. Table 3 shows the size of the final seeds set obtained in our model and the influence spread by each of the considered algorithms for both datasets. Remind that the influence spread is the number of influenced nodes with the computed influential set. At this stage, we should remind that a major advantage of our model SAIM is that it does not require the seed size as input. Instead, the latter is computed by Algorithm 3. On the opposite the rest of the considered approaches (High, CELF++, CDH, SMG, IMM, and CINEMA) require the seed size as input. To deal

Table 3. Final Seeds Size (*#seeds*) and the Influence Spread (IS) for Real Social Networks

Algorithm	Tencent Weibo		Higgs Tweet	
	#seeds	IS	*#seeds*	IS
High	124	1 859 645	138	408 311
CELF++	106	1 936 733	126	416 927
CDH	97	1 997 241	103	438 945
SMG	114	2 194 057	86	443 709
CINEMA	359	2 260 955	263	449 718
IMM	148	2 068 514	121	437 503
SAIM	**81**	**2 307 451**	**63**	**456 188**

②Kddcup. http://www.kddcup2012.org/c/kddcup2012-track1, Jan. 2018.

③Snap. https://snap.stanford.edu/data/higgs-twitter.html, Jan. 2018.

with this issue, for any given seed size s (varying from 5 to 50), we considered only the top-s seeds computed by SAIM according to their influence power. Table 4 and Table 5 report the influence spread of the computed influential set by varying the number of seeds from 5 to 50. Naturally, the larger the number, the better the algorithm. The first column in both tables reports the number of seed nodes. It is clear that our algorithm SAIM is able to compute the highest influence spread for both networks. The closest model to SAIM is CINEMA (the 6th column in both tables). Undoubtedly, the reasons can be attributed to the importance of social actions between users. We should notice that CINEMA[31] computes the influence by leveraging the algorithm CASINO which labels the edges by the signs (positive or negative) based on 5-level sentiment (like, somewhat like, neutral, somewhat dislike, dislike). However, our model SAIM does not treat social actions equally but rather weights each of them according to their importance (the level of influence). This explains somewhat why SAIM outperforms CINEMA. In order to get further insights on the pruning capabilities of SAIM, we considered the number of black (or significant) nodes, the maximal and the minimal length of the computed black paths④. These measures are reported in Table 6 and Table 7 for both datasets, where $\#seeds$ is the number of seeds, $\#SN$ is the number of significant nodes, MXB is the maximal length of black paths, and MNB is the minimal length of black paths. For example, if we consider the Tencent Weibo dataset, we can see in the first row of Table 6 that our model SAIM selects an influential set of size 5 out of 7 346 black (significant) nodes, and that it selects a single black seed node among a black path composed of 1 863 nodes. If we define the pruning rate as the number of pruned nodes ($\#SN - \#seeds$) divided by the number of significant ones ($\#SN$), then it is easy to see that this pruning rate is above 99%.

In order to visualize the influence power of the computed seed nodes, we extracted 115 509 nodes and 164 713 edges from the Tencent Weibo social network, whereas we extracted 19 548 nodes and 22 977 edges from Higgs Tweet social network. Plotting the whole social graphs is inadequate. Thereafter, we visualize these extracted parts as the "influence propagation map" (IPM) depicted in Fig.7. In this map, we visualize the influence power and the influence spread of the nodes. In Fig.7, the node size and the color gradient are

Table 4. Influence Spread on Tencent Weibo Dataset

$\#seeds$	High	CELF++	CDH	SMG	CINEMA	IMM	SAIM
5	518 741	520 773	533 291	542 640	544 211	539 784	**544 901**
10	734 807	738 921	747 112	750 243	757 152	751 392	**758 643**
15	890 711	899 674	902 022	902 065	907 614	904 871	**907 737**
20	989 709	1 001 588	1 005 317	1 018 723	1 034 337	1 027 162	**1 036 582**
25	1 100 862	1 107 410	1 112 702	1 138 143	1 149 266	1 146 418	**1 150 796**
30	1 183 544	1 196 157	1 243 339	1 244 020	1 244 483	1 244 211	**1 244 605**
35	1 254 038	1 272 513	1 299 076	1 313 524	1 317 927	1 319 386	**1 324 528**
40	1 293 041	1 337 368	1 380 708	1 383 713	1 387 092	1 385 015	**1 390 656**
45	1 340 218	1 395 455	1 410 129	1 426 770	1 436 851	1 427 939	**1 445 623**
50	1 382 844	1 452 363	1 468 203	1 471 641	1 482 309	1 473 561	**1 487 089**

Table 5. Influence Spread on Higgs Tweet Dataset

$\#seeds$	High	CELF++	CDH	SMG	CINEMA	IMM	SAIM
5	63 097	63 814	64 014	64 212	64 311	64 293	**64 335**
10	101 235	102 781	104 541	105 393	106 601	105 212	**106 745**
15	138 872	140 002	142 282	145 036	147 932	143 947	**148 557**
20	174 011	175 048	177 099	181 557	183 627	179 906	**187 480**
25	205 784	207 833	208 762	213 038	219 378	216 521	**221 995**
30	233 779	235 325	237 783	242 616	245 063	242 987	**249 814**
35	258 133	259 887	263 961	268 219	271 938	268 314	**275 292**
40	280 417	281 232	288 759	292 156	296 344	293 775	**298 971**
45	298 509	301 451	309 301	312 302	315 079	313 479	**317 763**
50	315 806	317 009	326 106	329 648	329 966	327 947	**330 448**

④We should remind that the length of a black path is its number of edges.

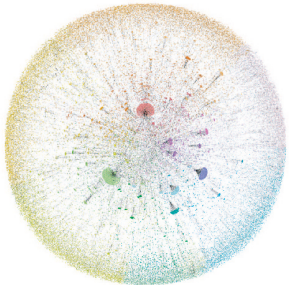
Table 6. Pruning Capability on Tencent Weibo Dataset

#seeds	*#SN*	*MXB*	*MNB*	Pruning Rate (%)
5	7 346	1 863	834	99.93
10	11 412	1 057	569	99.91
15	14 079	609	491	99.89
20	15 931	402	329	99.87
25	17 948	468	318	99.86
30	19 544	373	271	99.84
35	20 938	337	211	99.83
40	22 099	287	189	99.81
45	23 297	307	181	99.80
50	24 042	179	125	99.79

Table 7. Pruning Capability on Higgs Tweet Dataset

#seeds	*#SN*	*MXB*	*MNB*	Pruning Rate (%)
5	6 025	2 124	667	99.91
10	11 143	1 231	612	99.91
15	14 250	729	478	99.89
20	16 865	836	363	99.88
25	19 046	592	428	99.86
30	20 756	649	205	99.85
35	22 319	593	351	99.84
40	23 593	706	488	99.83
45	25 299	815	374	99.82
50	26 267	417	283	99.80

(a)



(b)

Fig.7. SAIM influence propagation shown for (a) Higgs Tweet and (b) Tencent Weibo social networks.

proportional to the influence power value (i.e., personalized PageRank centrality), and the network structure⑤ is presented using the algorithm proposed by Hu[64]. For instance, in Fig.7(a) a focal node is represented by a dark green color and will be degraded to light green in its neighbors. Now, we are ready to make the following main observations on this IPM. First, we remark that we have few focal nodes (with high influence power values) whose influence is spread through the network following friends and "friends-of-friends" links. For example, let us consider the IPM for the Higgs Tweet network plotted in Fig.7(a). We can observe three prominent leaders whose influence spreads following "friends-of-friends" paths until this influence is no longer significant. Clearly, the "communities" influenced by each seed are highlighted in distinct colors that are gradient to transparency to explain the proportion of direct and indirect influence. In fact, we can observe in Fig.7(a) that this color decays as we move from a focal node to its neighbors, i.e., as the distance from the focal nodes augments. The same remarks apply to the Tencent Weibo network in Fig.7(b) where in addition we have observed that 38% of the nodes receive information from the focal nodes with a distance less than 10 edges.

As a final criteria, we considered the running time of all algorithms averaged over 50 trials for each dataset, i.e., using seed number varying from 1 to 50. CELF++ and SMG as suggested by their authors[61-62] were simulated with Monte Carlo simulation using $R = 10\,000$ samples. The running time (in minutes) is reported in Table 8. We should remark that the computation time for CINEMA is reported for the l-way partitioning algorithm because the partition using the BFS technique takes much longer. We can see Table 8 that our model SAIM is slower than CELF++, SMG, CDH and IMM, and is faster than CINEMA. This is mainly due to the costly computing time of PageRank⑥.

Table 8. Average Running Time (min)

Algorithm	Tencent Weibo	Higgs Tweet
CELF++	201.54	193.42
SMG	10.61	7.08
CDH	27.28	25.38
CINEMA	726.43	401.54
IMM	38.91	29.18
SAIM	494.20	241.34

⑤This figure is spotted using Gephi[63] and its electronic version can be zoomed in for clarity using any PDF viewer.

⑥We refer here to the complexity analysis of the algorithm where it is clear that the PPR step is the most expensive.

4.2 Evaluation on Artificial Networks

In the second set of runs, we used synthetic datasets generated by the standard LFR[64] benchmarks. At this stage, we should remark that LFR generator has several parameters. Among these parameters, let us mention the number of nodes (N), the average degree of incoming edges (k), and the maximum degree of the incoming edges ($maxk$). For this, we designed two experiments in order to deal with distinct network complexities. In both experiments, we generated random social graphs along with their social actions of 5 000 nodes. Similarly, the random 5-level sentiment was generated to simulate CINEMA and CASINO algorithms. However, in the first experiment, we considered nodes with low degrees ($k = 20$ and $maxk = 50$), whereas in the second experiment, we considered nodes with high degrees ($k = 50$ and $maxk = 80$). Simulation results regarding the spreading criteria for both experiments are reported in Table 9. It is clear that CDH gives the best results for low-degree networks and that SAIM is not so far from it. One can observe that the number of seeds computed by SAIM is the lowest, but still having a significant influence spread especially for high degree networks where SAIM reached the highest spread with the lowest number of seeds. Naturally, this can be seen as a good indicator for performance. We can also notice in Table 9 that SMG, CDH and our model SAIM tend to give better results for high-degree networks. This seems to be "natural" since in high degree networks, influential nodes tend to have high connectivity and thereby are able to spread information much faster.

Table 9. Final Seeds (*#seeds*) Set and Influence Spread (IS) for Synthetic Datasets with Low and High Degrees

Algorithm	Low Degree Network		High Degree Network	
	#seeds	IS	*#seeds*	IS
High	113	3 477	137	3 592
CELF++	102	3 705	128	3 873
CDH	**74**	**4 841**	89	4 021
SMG	87	4 328	68	4 366
IMM	112	4 577	139	4 537
CINEMA	157	4 801	124	4 783
SAIM	63	4 673	**56**	**4 815**

As a conclusion according to these extensive experi mental results on both synthetic and real-world social networks, we can say that our proposal SAIM gives good performance compared with the state-of-the-art models. Indeed, SAIM ensures a large spread using the minimal number of seeds. This clearly shows the effectiveness of "social action" based models for influence maximization.

5 Conclusions

In this paper, we presented a new model called SAIM for the influence maximization (IM) problem in social networks. In SAIM, the IM problem is perceived as a social-actions influence maximization problem and focuses on the introduction of the activities of members to identify the most active users in the network. Moreover, SAIM assigns weights to social actions reflecting their levels of influence. SAIM is mainly composed of two major steps: 1) computation of influence power, and 2) selection of influential users. The main goal of the first phase is to define a new measure based on the social action that aims at computing the influence score for each user. A key feature of our model is the distinction between social actions that an individual can receive. In the second phase, we compute the set of influential nodes using a new concept called "influence-BFS tree". Hence, the most influential nodes are those having the influence-BFS trees that ensure the fastest spread of information. An experimental comparison of our model with the recent models reveals its good performance in computing optimal influential seeds on both real and synthetic social networks. As any research, the current work could be enhanced in several ways. Our immediate concern is to cast SAIM within a Hadoop/MapReduce framework in order to deal with social networks composed of billions of nodes.

References

[1] Farhadi F, Sorkhi M, Hashemi S, Hamzeh A. An effective framework for fast expert mining in collaboration networks: A group-oriented and cost-based method. *Journal of Computer Science and Technology*, 2012, 27(3): 577-590.

[2] Bouguessa M, Ben Romdhane L. Identifying authorities in online communities. *ACM Trans. Intelligent Systems and Technology*, 2015, 6(3): Article No. 30.

[3] Lv L Y, Zhang Y C, Yeung C H, Zhou T. Leaders in social networks, the *Delicious* case. *PLoS One*, 2011, 6(6): Article No. e21202.

[4] Zhang B L, Qian Z Z, Li W Z, Tang B, Lu S L, Fu X M. Budget allocation for maximizing viral advertising in social networks. *Journal of Computer Science and Technology*, 2016, 31(4): 759-775.

[5] Chen W, Li F, Lin T, Rubinstein A. Combining traditional marketing and viral marketing with amphibious influence maximization. In *Proc. the 16th ACM Conf. Economics and Computation*, June 2015, pp.779-796.

[6] Sangachin M, Samadi M, Cavuoto L. Modeling the spread of an obesity intervention through a social network. *Journal of Healthcare Engineering*, 2014, 5(3): 293-312.

[7] Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N. Cost-effective outbreak detection in networks. In *Proc. the 13th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, August 2007, pp.420-429.

[8] Kempe D, Kleinberg J, Tardos E. Maximizing the spread of influence through a social network. In *Proc. the 9th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, August 2003, pp.137-146.

[9] Domingos P, Richardson M. Mining the network value of customers. In *Proc. the 7th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, August 2001, pp.57-66.

[10] Chen W, Yuan Y F, Zhang L. Scalable influence maximization in social networks under the linear threshold model. In *Proc. the 10th IEEE Int. Conf. Data Mining*, December 2010, pp.88-97.

[11] Jung K, Heo W, Chen W. IRIE: Scalable and robust influence maximization in social networks. In *Proc. the 12th Int. Conf. Data Mining*, December 2012, pp.918-923.

[12] Wang C, Chen W, Wang Y J. Scalable influence maximization for independent cascade model in large-scale social networks. *Data Mining and Knowledge Discovery*, 2012, 25(3): 545-576.

[13] Kim J, Kim S K, Yu H. Scalable and parallelizable processing of influence maximization for large-scale social networks? In *Proc. the 29th Int. Conf. Data Engineering*, April 2013, pp.266-277.

[14] Wang Q Y, Jin Y H, Lin Z, Cheng S D, Yang T. Influence maximization in social networks under an independent cascade-based model. *Physica A: Statistical Mechanics and its Applications*, 2016, 444: 20-34.

[15] Bozorgi A, Haghighi H, Zahedi M S, Rezvani M. INCIM: A community-based algorithm for influence maximization problem under the linear threshold model. *Information Processing & Management*, 2016, 52(6): 1188-1199.

[16] Goyal A, Lu W, Lakshmanan L V S. SIMPATH: An efficient algorithm for influence maximization under the linear threshold model. In *Proc. the 11th Int. Conf. Data Mining*, December 2011, pp.211-220.

[17] Rahimkhani K, Aleahmad A, Rahgozar M, Moeini A. A fast algorithm for finding most influential people based on the linear threshold model. *Expert Systems with Applications*, 2015, 42(3): 1353-1361.

[18] Wang Y, Cong G, Song G J, Xie K Q. Community-based greedy algorithm for mining top-K influential nodes in mobile social networks. In *Proc. the 16th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, July 2010, pp.1039-1048.

[19] Jiang Q Y, Song G J, Cong G, Wang Y, Si W J, Xie K Q. Simulated annealing based influence maximization in social networks. In *Proc. the 25th AAAI Conf. Artificial Intelligence*, August 2011, pp.127-132.

[20] Tang Y Z, Shi Y C, Xiao X K. Influence maximization in near-linear time: A martingale approach. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, May 31-June 4, 2015, pp.1539-1554.

[21] Li H, Cui J T, Ma J F. Social influence study in online networks: A three-level review. *Journal of Computer Science and Technology*, 2015, 30(1): 184-199.

[22] Riquelme F, González-Cantergiani P. Measuring user influence on Twitter: A survey. *Information Processing & Management*, 2016, 52(5): 949-975.

[23] Tejaswi V, Bindu P V, Thilagam P S. Diffusion models and approaches for influence maximization in social networks. In *Proc. Int. Conf. Advances in Computing Communications and Informatics*, September 2016, pp.1345-1351.

[24] Weng J S, Lim E P, Jiang J, He Q. TwitterRank: Finding topic-sensitive influential twitterers. In *Proc. the 3rd ACM Int. Conf. Web Search and Data Mining*, February 2010, pp.261-270.

[25] Barbieri N, Bonchi F, Manco G. Topic-aware social influence propagation models. *Knowledge and Information Systems*, 2013, 37(3): 555-584.

[26] Xiang B, Liu Q, Chen E H, Xiong H, Zheng Y, Yang Y. PageRank with priors: An influence propagation perspective. In *Proc. the 23rd Int. Joint Conf. Artificial Intelligence*, August 2013, pp.2740-2746.

[27] Wang Y F, Vasilakos A V, Jin Q, Ma J H. PPRank: Economically selecting initial users for influence maximization in social networks. *IEEE Systems Journal*, 2017, 11(4): 2279-2290.

[28] Wang G J, Jiang W J, Wu J, Xiong Z L. Fine-grained feature-based social influence evaluation in online social networks. *IEEE Trans. Parallel and Distributed Systems*, 2014, 25(9): 2286-2296.

[29] Chen Y C, Chang S H, Chou C L, Peng W C, Lee S Y. Exploring community structures for influence maximization in social networks. In *Proc. the 6th SNA-KDD Workshop*, August 2012.

[30] Kandhway K, Kuri J. Using node centrality and optimal control to maximize information diffusion in social networks. *IEEE Trans. Systems Man and Cybernetics: Systems*, 2017, 47(7): 1099-1110.

[31] Li H, Bhowmick S S, Sun A X, Cui J T. Conformity-aware influence maximization in online social networks. *The VLDB Journal*, 2015, 24(1): 117-141.

[32] Li H, Bhowmick S S, Sun A X. CASINO: Towards conformity-aware social influence analysis in online social networks. In *Proc. the 20th ACM Int. Conf. Information and Knowledge Management*, October 2011, pp.1007-1012.

[33] Li Y H, Chen W, Wang Y J, Zhang Z L. Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships. In *Proc. the 6th ACM Int. Conf. Web Search and Data Mining*, February 2013, pp.657-666.

[34] He J, Kaur H, Talluri M. Positive opinion influential node set selection for social networks: Considering both positive and negative relationships. In *Proc. Wireless Communications Networking and Applications*, December 2014, pp.935-948.

[35] Guler B, Varan B, Tutuncuoglu K, Nafea M, Zewail A A, Yener A, Octeau D. Using social sensors for influence propagation in networks with positive and negative relationships. *IEEE Journal of Selected Topics in Signal Processing*, 2015, 9(2): 360-373.

[36] Liu S Y, Wang S H, Zhu F D, Zhang J B, Krishnan R. HYDRA: Large-scale social identity linkage via heterogeneous behavior modeling. In *Proc. ACM SIGMOD Int. Conf. Management of Data*, June 2014, pp.51-62.

[37] Liu S Y, Wang S H, Zhu F D. Structured learning from heterogeneous behavior for social identity linkage. *IEEE Trans. Knowledge and Data Engineering*, 2015, 27(7): 2005-2019.

[38] Subbian K, Sharma D, Wen Z, Srivastava J. Finding influencers in networks using social capital. In *Proc. IEEE/ACM Int. Conf. Advances in Social Networks Analysis and Mining*, August 2013, pp.592-599.

[39] Franks H, Griffiths N, Anand S S. Learning influence in complex social networks. In *Proc. Int. Conf. Autonomous Agents and Multi-agent Systems*, May 2013, pp.447-454.

[40] Deng X H, Pan Y, Wu Y, Gui J S. Credit distribution and influence maximization in online social networks using node features. In *Proc. the 12th Int. Conf. Fuzzy Systems and Knowledge Discovery*, August 2015, pp.2093-2100.

[41] Liu G F, Zhu F, Zheng K, Liu A, Li Z X, Zhao L, Zhou X F. TOSI: A trust-oriented social influence evaluation method in contextual social networks. *Neurocomputing*, 2016, 210: 130-140.

[42] Zeng Y F, Chen X F, Cong G, Qin S C, Tang J, Xiang Y P. Maximizing influence under influence loss constraint in social networks. *Expert Systems with Applications*, 2016, 55: 255-267.

[43] Subbian K, Aggarwal C, Srivastava J. Mining influencers using information flows in social streams. *ACM Trans. Knowledge Discovery from Data*, 2016, 10(3): Article No. 26.

[44] Liu S Y, Chen L, Ni L M, Fan J P. CIM: Categorical influence maximization. In *Proc. the 5th Int. Conf. Ubiquitous Information Management and Communication*, February 2011, Article No. 124.

[45] Qu Q, Liu S Y, Jensen C S, Zhu F D, Faloutsos C. Interestingness-driven diffusion process summarization in dynamic networks. In *Proc. European Conf. Machine Learning and Knowledge Discovery in Databases*, September 2014, pp.597-613.

[46] On B W, Lim E P, Jiang J, Teow L N. Engagingness and responsiveness behavior models on the Enron email network and its application to email reply order prediction. In *The Influence of Technology on Social Network Analysis and Mining*, Özyer T, Rokne J, Wagner G, Reuser A H P (eds.), Springer, 2013, pp.227-253.

[47] Achananuparp P, Lim E P, Jiang J, Hoang T A. Who is retweeting the tweeters? Modeling, originating, and promoting behaviors in the Twitter network. *ACM Trans. Management Information Systems*, 2012, 3(3): Article No. 13.

[48] Zhao K, Yen J, Greer G, Qiu B J, Mitra P, Portier K. Finding influential users of online health communities: A new metric based on sentiment influence. *Journal of the American Medical Informatics Association*, 2014, 21(e2): e212-e218.

[49] Yang C C, Tang X N. Estimating user influence in the MedHelp social network. *IEEE Intelligent Systems*, 2012, 27(5): 44-50.

[50] Nikolaev A, Gore S, Govindaraju V. Engagement capacity and engaging team formation for reach maximization of online social media platforms. In *Proc. the 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, August 2016, pp.225-234.

[51] Bonchi F, Castillo C, Gionis A, Jaimes A. Social network analysis and mining for business applications. *ACM Trans. Intelligent Systems and Technology*, 2011, 2(3): Article No. 22.

[52] Fang Q, Sang J T, Xu C S, Rui Y. Topic-sensitive influencer mining in interest-based social media networks via hypergraph learning. *IEEE Trans. Multimedia*, 2014, 16(3): 796-812.

[53] Li J X, Liu C F, Yu J X, Chen Y, Sellis T, Culpepper J S. Personalized influential topic search via social network summarization. *IEEE Trans. Knowledge and Data Engineering*, 2016, 28(7): 1820-1834.

[54] Chen Y C, Zhu W Y, Peng W C, Lee W C, Lee S Y. CIM: Community-based influence maximization in social networks. *ACM Trans. Intelligent Systems and Technology*, 2014, 5(2): Article No. 25.

[55] Budak C, Agrawal D, Abbadi A E. Limiting the spread of misinformation in social networks. In *Proc. the 20th Int. Conf. World Wide Web*, April 2011, pp.665-674.

[56] Al-Garadi M A, Varathan K D, Ravana S D. Identification of influential spreaders in online social networks using interaction weighted K-core decomposition method. *Physica A: Statistical Mechanics and its Applications*, 2017, 468: 278-288.

[57] Chen W L, Cheng S Y, He X, Jiang F. InfluenceRank: An efficient social influence measurement for millions of users in microblog. In *Proc. the 2nd Int. Conf. Cloud and Green Computing*, November 2012, pp.563-570.

[58] Brin S, Page L. The anatomy of a large-scale hypertextual web search engine. In *Proc. the 7th Int. World-Wide Web Conf.*, April 1998, pp.107-117.

[59] Lee S, Park S, Kahng M, Lee S G. PathRank: Ranking nodes on a heterogeneous graph for flexible hybrid recommender systems. *Expert Systems with Applications*, 2013, 40(2): 684-697.

[60] Boyd S, Vandenberghe L. Convex Optimization (7th edition). Cambridge University Press, 2009.

[61] Goyal A, Lu W, Lakshmanan L V S. CELF++: Optimizing the greedy algorithm for influence maximization in social networks. In *Proc. the 20th Int. Conf. Companion on World Wide Web*, March 2011, pp.47-48.

[62] Heidari M, Asadpour M, Faili H. SMG: Fast scalable greedy algorithm for influence maximization in social networks. *Physica A: Statistical Mechanics and its Applications*, 2015, 420: 124-133.

[63] Hu Y F. Efficient, high-quality force-directed graph drawing. *The Mathematica Journal*, 2006, 10(1): 37-71.

[64] Bastian M, Heymann S, Jacomy M. Gephi: An open source software for exploring and manipulating networks. In *Proc. the 3rd Int. AAAI Conf. Weblogs and Social Media* , July 2009, pp.361-362.

[65] Lancichinetti A, Fortunato S, Radicchi F. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 2008, 78(4): Article No. 046110.

Mehdi Azaouzi is currently a Ph.D. student at the National School of Computer Sciences, University of Manouba, Tunisia. He received his Bachelor's degree and his Master's degree from the Faculty of Sciences, University of Monastir, Tunisia, both in computer science, in 2009 and 2012, respectively. His current research interests include social networks analyses, graph mining, and graph indexing. He is member of the MARS (Modeling of Automated Reasoning Systems) Research Laboratory.

Lotfi Ben Romdhane received his Ph.D. degree from the University of Sherbrooke, Quebec, and an engineering degree from National School of Computer Sciences (ENSI), University of Manouba, Tunisia, both in computer science, in 1994 and 2000 respectively. He is currently a professor in computer science at Higher Institute of Computer Science and Telecom (ISITCom), University of Sousse, Tunisia, and heads MARS (Modeling of Automated Reasoning Systems) Research Laboratory and the SDM (Smart Data Mining) research group. He has conducted several joint research projects in the field of social networks analyses with LIP6, CNAM/France, and with the University of Quebec at Montreal (UQAM), Quebec . His areas of expertise span the general area of Data Science and include reasoning, distributed computing, knowledge discovery, and data mining. He has published more than 60 papers in these topics in international conferences and journals.